UNIVERSAL COMPUTABLY ENUMERABLE SETS AND INITIAL SEGMENT PREFIX-FREE COMPLEXITY

GEORGE BARMPALIAS

Abstract. We show that there are Turing complete computably enumerable sets of arbitrarily low non-trivial initial segment prefix-free complexity. In particular, given any computably enumerable set A with non-trivial prefix-free initial segment complexity, there exists a Turing complete computably enumerable set B with complexity strictly less than the complexity of A. On the other hand it is known that sets with trivial initial segment prefix-free complexity are not Turing complete.

Moreover we give a generalization of this result for any finite collection of computably enumerable sets $A_i, i < k$ with non-trivial initial segment prefix-free complexity. An application of this gives a negative answer to a question from [DH10, Section 11.12] and [MS07] which asked for minimal pairs in the structure of the c.e. reals ordered by their initial segment prefix-free complexity.

Further consequences concern various notions of degrees of randomness. For example, the Solovay degrees and the K-degrees of computably enumerable reals and computably enumerable sets are not elementarily equivalent. Also, the degrees of randomness based on plain and prefix-free complexity are not elementarily equivalent; the same holds for their Δ^0_2 and Σ^0_1 substructures.

1. Introduction

The interplay between the information that can be coded into an infinite binary sequence and its initial segment complexity has been the subject of a lot of research in the last ten years. A rather influential result from [DHNS03] that spawned a renewed interest in this area was that sequences with very easily describable initial segments cannot compute the halting problem. Moreover the method that was used to establish it, often referred to as the decanter method, was novel and inspired much of the deeper work in this area. We show that although a universal computably enumerable set does not have trivial initial segment complexity, it can have arbitrarily low non-trivial initial segment complexity. Moreover our method is dual to the decanter method and in this sense the present paper can be seen as a missing companion to [DHNS03].

We start with a brief overview of Kolmogorov complexity in Section 1.1 and measures of relative randomness in Section 1.2 with a special attention to the topics around our results. In Section 1.3 we discuss the class of sequences with trivial initial segment complexity along with the motivation of our results, which

First version: 1 October 2012. *Current version:* 2 November 2012.

Key words and phrases. Universal sets, computably enumerable, Kolmogorov complexity, initial segment complexity.

Barmpalias was supported by a research fund for international young scientists No. 611501-10168 and an *International Young Scientist Fellowship* number 2010-Y2GB03 from the Chinese Academy of Sciences. Partial support was also obtained by the *Grand project: Network Algorithms and Digital Information* of the Institute of Software, Chinese Academy of Sciences.

are presented in Section 1.4. A number of applications are given in Section 1.5 and Section 1.6 discusses connections of the present work with research on other reducibilities that are related to Kolmogorov complexity. In Section 2 we introduce the main technical tools that are required for the proofs of our results and Sections 3 and 4 contain the proofs of the two main results respectively.

1.1. **Kolmogorov complexity and randomness.** A standard measure of the complexity of a finite string was introduced by Kolmogorov in [Kol65]. The basic idea behind this approach is that simple strings have short descriptions relative to their length while complex or random strings are hard to describe concisely. Kolmogorov formalized this idea using the theory of computation. In this context, Turing machines play the role of our idealized computing devices, and we assume that there are Turing machines capable of simulating any mechanical process which proceeds in a precisely defined and algorithmic manner. Programs can be identified with binary strings. A string τ is said to be a description of a string σ with respect to a Turing machine M if this machine halts when given program τ and outputs σ. Then the Kolmogorov complexity of σ with respect to M (denoted by $C_M(\sigma)$) is the length of its shortest description with respect to M. It can be shown that there exists an *optimal* prefix-free machine V, i.e. a machine which gives optimal complexity for all strings, up to a certain constant number of bits. This means that for each Turing machine M there exists a constant c such that $C_V(\sigma) < C_M(\sigma) + c$ for all finite strings σ.

When we come to consider randomness for infinite strings, it becomes important to consider machines whose domain satisfies a certain condition; the machine M is called *prefix-free* if it has prefix-free domain (which means that no program for which the machine halts and gives output is an initial segment of another). The complexity of a string σ with respect to a prefix-free machine M is denoted by $K_M(\sigma)$. As with the case of plain Turing machines, there exists an *optimal* prefix-free machine U. This means that for each prefix-free machine M there exists a constant c such that $K_U(\sigma) < K_M(\sigma) + c$ for all finite strings σ.

According to the above discussion, both in the case of plain or prefix-free Turing machines the choice of the underlying optimal machine does not change the complexity distribution significantly. Hence the theories of plain and prefix-free complexity can be developed without loss of generality, based on fixed underlying optimal plain and prefix-free machines V, U. We let $C = C_V$ and $K = K_U$.

In order to define randomness for infinite sequences, we consider the complexity of all finite initial segments. A finite string σ is said to be *c-incompressible* if $K(\sigma) \geq |\sigma| - c$. Levin [Lev73] and Chaitin [Cha75] defined an infinite binary sequence X to be random if there exists some constant c such that all of its initial segments are c-incompressible. By identifying subsets of $\mathbb{N}$ with their characteristic sequence we can also talk about randomness of sets of numbers. This definition of randomness of infinite sequences is independent of the choice of underlying optimal prefix-free machine, and coincides with other definitions of randomness like the definition given by Martin-Löf in [ML66]. The coincidence of the randomness notions resulting from various different approaches may be seen as evidence of a robust and natural theory.

1.2. **Measures of relative randomness.** Once a solid definition of initial segment complexity and randomness is in place, it is often desirable to have a way to compare two infinite binary sequences in this respect. One of the early measures of

relative initial segment complexity was developed by Solovay in [Sol75] especially for the computably enumerable (c.e.) reals. These are binary expansions of the real numbers in the unit interval which are limits of increasing computable sequences of rationals. The Solovay reducibility gave a formal way to compare c.e. reals with respect to the difficulty of getting good approximations to them. Solovay showed in [Sol75] that the induced degree structure has a complete element which contains exactly the random c.e. reals. The Solovay degrees of c.e. reals where further studied in [DHN02, DHL07] (see [DH10, Section 9.5] for an overview).

Downey, Hirschfeld and LaForte [DHL04] introduced and studied a number of other measures of relative initial segment complexity that are not restricted to the c.e. reals. Most of them are extensions of the Solovay measure of relative complexity. For example, they defined $A \leq_K B$ if $\exists c \forall n\ (K(A\restriction_n) \leq K(B\restriction_n) + c)$; in other words, if the prefix-free complexity of each initial segment of A is bounded by the prefix-free complexity of the corresponding initial segment of B, modulo a constant. This reducibility, already implicit in [Sol75], is a proper extension of the Solovay reducibility on the c.e. reals and was further studied in [YDD04, MY08, MY10] with a special attention to random sequences and in [CM06, MS07, BV11], [Bar11, Section 5] with more focus on local properties. A lot of these results refer to the degree structure that is induced by $\leq_K$, the K-degrees. A version of $\leq_K$ for plain Kolmogorov complexity was also defined in [DHL04], which induces the structure of the C-degrees. In particular, $A \leq_C B$ if $\exists d \forall n\ (C(A\restriction_n) \leq C(B\restriction_n) + d)$.

1.3. **Trivial initial segment complexity and Turing degrees.** A string σ that has prefix-free complexity as low as the prefix-free complexity of the sequence of 0s of the same length may be regarded as trivial. Indeed, if we consider the prefix-free complexity of a string as a measure of the information that is coded in the string, in this case there is no information coded in the bits of the sequence. The infinite sequences whose initial segments have trivial prefix-free complexity are known as the K-trivial sequences. Formally, X is K-trivial if $\exists c \forall n\ (K(A\restriction_n) \leq K(n) + c)$, where we may identify $K(n)$ with $K(0^n)$. Surprisingly, there are noncomputable K-trivial sequences and this was already proved in [Sol75]. Note that the K-trivial sequences are the contents of the least element in the K-degrees that were discussed in Section 1.2.

An interesting question that motivated a lot of later research was the following.

(1.1) How much information can be encoded in an infinite binary sequence with very simple initial segments?

In particular, is it possible to encode a Turing complete problem into a K-trivial sequence. A particularly simple construction of a noncomputable K-trivial c.e. set that was presented in [DHNS03] made this possibility plausible. However in the same paper it was shown that this is not the case. In particular, if an oracle A computes the halting problem then for each constant c there are initial segments σ of A such that $K(\sigma) > K(|\sigma|) + c$. The proof of this result was quite novel, and along with its extensions it became known as *the decanter method*. Hirschfeldt and Nies extended this method in [Nie05] and showed that the amount of information that can be coded into K-trivial sequences is in fact quite limited. Quite interestingly, they also showed that K-triviality is downward closed with respect to Turing reductions. We refer to [DH10, Section 11.4] and [Nie09, Section 5] for detailed presentations of the decanter method.

1.4. **Motivation and results.** In this paper we revisit question (1.1) by examining the possibility of coding considerable information in an infinite sequence with initial segments of very low but not necessarily trivial prefix-free complexity. We initially focus in the special case of c.e. sets, where Turing completeness provides a notion of maximality of information that can be coded. Hence we may ask the following question.

(1.2) How low can be the initial segment prefix-free complexity of a Turing complete computably enumerable set?

How can we qualify the notion of 'low initial segment complexity' in question (1.2)? Note that modulo an additive constant, $K(n)$ is a lower bound on the complexity of the first n bits of any infinite sequence. Since the K-trivial sequences are ruled out by the result in [DHNS03], we turn our attention to sequences whose initial segment prefix-free complexity may deviate from the lower bound $K(n)$ but is still quite low. One way we could try to make this lowness condition precise is to look among sequences A such that $K(A\upharpoonright_n) - K(n)$ is bounded by a very slow growing function g, as it is shown in (1.3).

$$\exists c \forall n\ (K(A\upharpoonright_n) \leq K(n) + g(n) + c) \tag{1.3}$$

The notion of 'slow growing' may be quantified through the arithmetical hierarchy of complexity. For example there are Δ^0_3 unbounded nondecreasing functions that are dominated by all Δ^0_2 functions with the same properties. In this sense, as the rate of growth of a function is reduced (but remains nontrivial) the arithmetical complexity of it increases. Let us first consider nondecreasing functions g. In [BMN11, BB10] it was shown that if g is nondecreasing, unbounded and Δ^0_2 then there is a large uncountable collection of oracles A that satisfy (1.3). Hence a class that includes functions with these properties is not sufficiently restrictive for our purpose and we need to look in higher complexity classes. On the other hand in [CM06, BB10] it was shown that there are nondecreasing unbounded functions g in Δ^0_3 such that any set A that satisfies (1.3) is K-trivial. Moreover allowing functions that may decrease occasionally introduces similar problems. For example, it was shown in [BV11, Section 5] that there is a Δ^0_2 function g such that $\lim_n g(n) = \infty$ and any c.e. set which satisfies (1.3) is K-trivial. Hence condition (1.3) in combination with standard ways to quantify the rate of growth of the function g is not a fruitful way to formalize the notion of 'low nontrivial initial segment complexity'.

Another approach is to compare the initial segment complexity of a c.e. set with the complexity of other sets. Although this would not give us an absolute notion of low nontrivial complexity, an answer of the type 'lower than the complexity of any sequence with nontrivial complexity' to the question (1.2) would be definitive. The existence of minimal K-degrees is an open problem, but since this question refers to c.e. sets, such a strong positive answer is still not possible. Indeed, it was shown in [BV11] that there is a Δ^0_2 set B which is not K-trivial but every c.e. set with $A \leq_K B$ is K-trivial. In other words the initial segment complexity of B does not bound the complexity of any c.e. set with nontrivial initial segment complexity. This shows that the comparison needs to involve the complexities of c.e. sets and not arbitrary sequences. In this sense, the best possible answer to question (1.2) would be the existence of Turing complete c.e. sets with initial segment complexity strictly lower than the complexity of any given c.e. set that is not K-trivial. Our first result establishes exactly this.

Theorem 1.1. *Let A be a computably enumerable set which is not K-trivial. Then there exists a computably enumerable set B such that $B \equiv_T \emptyset'$ and $B <_K A$.*

The proof of Theorem 1.1 involves a very sparse coding of compete information, which produces a sequence with very simple initial segments, in the sense of the prefix-free complexity. A crucial part of the argument is the exploitation of the fact that the given set is c.e. and has nontrivial initial segment prefix-free complexity. In this sense Theorem 1.1 is dual to the main result of [DHNS03] that K-trivial sets are incomplete. More generally, the decanter method that was developed in [DHNS03] is a tool for exploiting the lack of complexity of a set in order to deduce additional properties. The method used in the proof of Theorem 1.1 is a tool for exploiting the complexity of a sequence (in combination with an effective approximation to it) in order to absorb the complexity of a coding procedure. In this sense the two methods are dual.

It is instructive to compare Theorem 1.1 with condition (1.3). If we wish to express our result in these terms we can set $g(n) = K(A \upharpoonright_n) - K(n)$. We note that $g(n)$ will be occasionally decreasing. In fact, it is well known that for every c.e. set A the $\liminf(K(A \upharpoonright_n) - K(n))$ is finite. In other words, c.e. sets are infinitely often K-trivial (see [BV11, Section 2] for a proof and a general discussion about infinitely often K-trivial sets). This observation gives some idea about the challenges of implementing the coding that is required in Theorem 1.1 as well as the qualification of the idea of 'low initial segment complexity' for c.e. sets.

Our second result is a generalization of Theorem 1.1 to any finite collection of c.e. sets with nontrivial initial segment prefix-free complexity. We state it and prove it for the special case of two c.e. sets since the more general version may be obtained trivially and effectively by an iterated application.

Theorem 1.2. *Let A, D be computably enumerable sets which are not K-trivial. Then there exists a computably enumerable set B such that $B <_K A$, $B <_K D$ and $B \equiv_T \emptyset'$.*

This extension has several applications that are discussed in Section 1.5, including the solution to an open question from [DH10, Section 11.12]. Moreover its proof goes considerably beyond a routine adaptation of the special case established in Theorem 1.1. As we elaborate in Section 4.2 the main obstacle is the lack of uniformity in the complexities of the given c.e. sets. This can be better understood if we recall that K-trivial sets are infinitely often K-trivial. In particular, as we discuss in Section 1.5, Theorem 1.2 shows that if two c.e. sets A, D are not K-trivial their initial segment complexity must rise simultaneously on some lengths. Hence despite the potential lack of uniformity in the oscilations of the complexity of two c.e. sets, there must be some uniformity on a local level i.e. places where the complexities $K(A \upharpoonright_n), K(D \upharpoonright_n)$ deviate from $K(n)$ simultaneously.

1.5. **Applications.** The first application concerns various local structures of the K-degrees. The existence of minimal pairs of K-degrees was established in [CM06], where two Δ^0_4 sets forming a minimal pair in this structure were constructed. In [MS07] a minimal pair of Σ^0_2 sets was presented and in [BV11, Section 3] it was shown that there is a Σ^0_2 set that forms a minimal pair with all Σ^0_1 sets in the K-degrees. Theorem 1.2 is equivalent to saying that there are no minimal pairs in the K-degrees of c.e. sets. In particular, there is no pair of Σ^0_1 sets that form

a minimal pair of K-degrees. This complements the existence results for minimal pairs in the K-degrees.

Downey and Hirschfeldt [DH10, Section 11.12] as well as Merkle and Stephan [MS07] asked if there is a pair of c.e. reals that form a minimal pair in the K-degrees. This question is particularly interesting since $\leq_K$ is often introduced as a generalization of the Solovay reducibility, which is the standard measure of relative randomness on the class of c.e. reals. We show that Theorem 1.2 answers this question in the negative. We need the following fact.

Lemma 1.3. *If A is a c.e. real such that $\emptyset <_K A$ then there exists a c.e. set B with $\emptyset <_K B \leq_K A$.*

Proof. Since A is a c.e. real, it has a computable approximation $(A[s])$ according to which if $A(n)[s] = 1$ and $A(n)[s+1] = 0$ then there is some $i < n$ such that $A(i)[s] = 0$ and $A(i)[s+1] = 1$. A canonical encoding of the approximation $(A[s])$ into a c.e. set B can be achieved based on the fact that for each n the value of $A(n)[s]$ can only change at most 2^n times during the stages s. The first bit of B encodes the oscillations to $A(0)$, the next 2 bits encode $A(1)$, the next 2^2 bits encode $A(2)$ and so on. In particular if $A(k)$ is encoded in the bits $(m, m+2^k]$ of B, upon each change in $A(k)[s]$ during the stages s we enumerate into B the largest element of $(m, m+2^k]$ that is not yet in B. In this way we have $A \equiv_T B$ and $B \leq_T A$ through a Turing reduction that uses at most n bits of A in the computation of n bits of B. Since K-triviality is a degree-theoretic property we have $\emptyset <_K B$ and by the basic properties of $\leq_K$ on the c.e. reals we also have $B \leq_K A$. □

By Theorem 1.2 and Lemma 1.3 we get the desired result about minimal pairs.

Corollary 1.4. *There are no minimal pairs in the K-degrees of c.e. reals.*

The separation of Solovay reducibility from $\leq_K$ on the c.e. reals was already achieved in [DHL04], where a pair of c.e. reals A, B was constructed such that $A \leq_K B$ but A is not Solovay reducible to B. However these examples are artificial since they were obtained via diagonalization. A more natural separation would be obtained by an elementary difference in the corresponding degree structures of c.e. reals. This is provided by the existence of minimal pairs which occurs in the Solovay degrees of c.e. reals by [DHL04] but not in the K-degrees of c.e. reals by Corollary 1.4. The same holds for c.e. sets according to Theorem 1.2.

Corollary 1.5. *The structures of the Solovay degrees and the K-degrees of computably enumerable reals are not elementarily equivalent. Moreover the same holds for the Solovay degrees and the K-degrees of computably enumerable sets.*

Merkle and Stephan showed in [MS07] that there exist two c.e. sets that from a minimal pair with respect to $\leq_C$. Hence Corollary 1.4 also provides an elementary difference between the C-degrees and the K-degrees as well as the local Δ^0_2 and Σ^0_1 degree structures.

Corollary 1.6. *The structures of the C-degrees and the K-degrees are not elementarily equivalent. Moreover the same holds for the corresponding Δ^0_2 and Σ^0_1 substructures.*

It would be interesting to find elementary differences between the K-degrees of c.e. reals and the K-degrees of c.e. sets. This was done in [Bar05] for the Solovay

degrees by showing that there are no maximal elements in the Solovay degrees of c.e. sets. We do not know the answer to the following question.

Are there maximal elements in the K-degrees of c.e. sets?

A related question would be if there is a complete degree in the K-degrees of c.e. sets.

A final application of Theorem 1.2 concerns the following question.

(1.4) Is there a pair of sequences X, Y which are not K-trivial but $\min\{K(X\restriction_n), K(Y\restriction_n)\} - K(n)$ has a constant upper bound?

Theorem 1.2 in combination with Lemma 1.3 answers (1.4) in the negative in the case where X, Y are required to be computably enumerable reals.

Corollary 1.7. *Suppose that A_i, $i < k$ is a finite collection of computably enumerable reals and each of them is not K-trivial. Then for all c there exists n such that $K(A_i\restriction_n) > K(n) + c$ for all $i < k$.*

We do not know the answer of (1.4) in general.

1.6. **Related work on weak reducibilities.** A method for exploiting the power of an oracle to achieve better compression of programs (along with a computable approximation to it) has been used in the study of another reducibility that is related to randomness and is called $\leq_{LK}$. We say that $X \leq_{LK} Y$ if $\exists c \forall \sigma\ (K^Y(\sigma) \leq K^X(\sigma) + c)$. In other words $X \leq_{LK} Y$ formalizes the notion that Y can achieve an overall compression of the strings that is at least as good as the compression achieved by X. Moreover by [KHMS12] it coincides with $X \leq_{LR} Y$ which denotes the relation that every random sequence relative to Y is also random relative to X. The degree structure that is induced by $X \leq_{LK} Y$ has a least element that turns out to contain exactly the K-trivial sequences. In [BM09] an argument was used that exploits the compression power of nontrivial c.e. sets in the study of the structure of c.e. sets under $\leq_{LK}$. A similar argument was used in [Bar10b] in order to show that every Δ^0_2 set with nontrivial compression power has uncountably many predecessors with respect to $\leq_{LK}$. In [Bar10a] this approach was further developed in order to exhibit elementary differences between various local structures of the LK degrees and the Turing degrees. We note that the arguments in these references work explicitly with $\leq_{LR}$ but can alternatively be implemented with the equivalent $\leq_{LK}$.

However there are some differences between $\leq_K$ and $\leq_{LK}$, the most important being that in $\leq_{LK}$ we usually work with oracle computations while in $\leq_K$ we only work with descriptions. It is quite remarkable that the triviality notion with respect to $\leq_K$ coincides with the triviality notion with respect to $\leq_{LK}$. A soon as we consider sequences of non-zero K-degrees or LK-degrees, the study of the two structures becomes less uniform. For example, it is instructive to compare the arguments about the non-existence of minimal pairs of K-degrees in this paper with the corresponding arguments in [Bar10a] that refer to the LK degrees.

2. Preliminaries

The main tool in the proof of these theorems is a method of coding information into a set B that is constructed, while keeping its initial segment complexity below the complexity of a given c.e. set A that is not K-trivial. It is a method for exploiting the fact that a given set has a computable enumeration and non-trivial

initial segment complexity, for the purpose of coding. In particular, it allows to meet the conflicting requirements $\emptyset' \leq_T B$ and $B \leq_K A$.

2.1. **Prefix-free machines.** For $B \leq_K A$ we need to build a prefix-free machine that witnesses the relation of the two complexities. Let U be the optimal prefix-free machine which underlies the prefix-free complexity K. Hence $K = K_U$. This machine is optimal in the sense that given any other prefix-free oracle machine M there is a constant c such that $K(\sigma) \leq K_M(\sigma) + c$ for all strings σ. The *weight* of a prefix-free set S of strings, denoted $\texttt{wgt}(S)$, is defined to be the sum $\sum_{\sigma\in S} 2^{-|\sigma|}$. The *weight* of a prefix-free machine M is defined to be the weight of its domain and is denoted $\texttt{wgt}(M)$. Without loss of generality we assume that $\texttt{wgt}(U) < 2^{-2}$.

Prefix-free machines are most often built in terms of *request sets.* A request set L is a set of tuples $\langle \rho, \ell\rangle$ where ρ is a string and ℓ is a positive integer. A 'request' $\langle \rho, \ell\rangle$ represents the intention of describing ρ with a string of length ℓ. We define the *weight of the request* $\langle \rho, \ell\rangle$ to be $2^{-\ell}$. We say that L is a *bounded request set* if the sum of the weights of the requests in L is less than 1. This sum is the *weight of the request set* L and is denoted by $\texttt{wgt}(L)$. The Kraft-Chaitin theorem (see e.g. [DH10, Section 2.6]) says that for every bounded request set L which is c.e., there exists a prefix-free machine M such that for each $\langle \rho, \ell\rangle \in L$ there exists a string τ of length ℓ such that $M(\tau) = \rho$. We freely use this method of construction without explicit reference to the Kraft-Chaitin theorem. A real number $0 \leq r < 1$ is called computably enumerable (c.e.) if it is the limit of a non-decreasing computable sequence of rational numbers. The binary strings are ordered first by length and then lexicographically.

2.2. **Coding.** The coding of $\emptyset'$ into B will be implemented through a system of movable markers $m_n, n \in \mathbb{N}$, where m_n represents the B-code of the possible enumeration of n into $\emptyset'$. The movement of the markers as well as the computable enumeration of B will take place in the stages of the enumeration of $\emptyset'$. In particular the value of m_n at stage s is denoted by $m_n[s]$. It is possible that $m_n[s]$ is undefined (in symbols, $m_n[s]\uparrow$) for some $n, s \in \mathbb{N}$. The movement of the markers satisfies the following standard properties:

(i) *Monotonicity on stages:* if $m_n[s]\downarrow, m_n[s+1]\downarrow$ then $m_n[s] < m_n[s+1]$;
(ii) *Monotonicity on indices:* if $m_n[s]\downarrow, m_{n+1}[s]\downarrow$ then $m_n[s] < m_{n+1}[s]$;
(iii) *Consistency:* if $m_n[s]\downarrow, m_n[t]\downarrow$, $m_n[s] \neq m_n[t]$ and $s < t$, then $m_n[s] \in B$;
(iv) *Convergence:* $\forall n\ \exists t, k\ \forall s\ (s > t \Rightarrow m_n[s]\downarrow = k)$;
(v) *Coding:* $\forall n\ (n \in \emptyset' \iff m_n \in B)$ where $m_n = \lim_s m_n[s]$.

Given a system of markers (m_n) with the above properties, we can compute $\emptyset'$ given B as follows. In order to decide if $n \in \emptyset'$ it suffices to use B to compute $m_n = \lim_s m_n[s]$ and then ask about the membership of m_n in B.

The essence of the method lies on the specific rules that determine the movement of the markers m_i. Intuitively, in order to maintain $B \leq_K A$ the markers are forced to move many times. Their convergence is a consequence of the failure to construct a machine demonstrating that A is K-trivial. Section 3 contains the formal argument.

It turns out that this type of sparse coding may be 'permitted' by any finite number of given c.e. sets that are not K-trivial. In particular, with some additional effort we can do the same coding into B while keeping its initial segment complexity below any two given c.e. sets A, C that are not K-trivial. Section 4 is devoted to the proof of this generalized result.

3. Proof of Theorem 1.1

Let A be a computably enumerable set which is not K-trivial. For the proof of Theorem 1.1 it suffices to construct a computably enumerable set B such that $B \equiv_T \emptyset'$ and $B \leq_K A$. This follows from the fact that the c.e. K-degrees are downward dense, i.e. for each c.e. set X such that $\emptyset <_K X$ there exists a c.e. set Y suc that $\emptyset <_K Y <_K X$; see [Bar11, Section 5].

In order to make B Turing complete we will use a system of markers (m_i) as we discussed in Section 2.2. In order to establish $B \leq_K A$ it suffices to construct a prefix-free machine M such that

$$K_M(B \restriction_n) \leq K(A \restriction_n) \quad \text{for all } n \tag{3.1}$$

where K_M denotes the prefix-free complexity relative to machine M. Recall that K denotes the prefix-free complexity relative to a fixed universal prefix-free machine U. Without loss of generality we may assume that $\texttt{wgt}(U) < 2^{-2}$. For each marker m_i we enumerate a prefix-free machine N_i during the construction. The purpose of N_i is to achieve $\forall n\ (K_{N_i}(A \restriction_n) \leq K(n) + c_i)$ for some constant c_i. Since A is not K-trivial, this will ultimately fail. However this failure will help demonstrate that m_i converges. The value of c_i may increase during the construction. This happens each time some m_j, $j < i$ moves. Such an event is often described as an 'injury' of m_i. In particular, if at some stage s marker m_k moves while m_j, $j < k$ remain constant this causes m_i, $i > k$ to be injured, which has the following consequences:

- m_i, $i > k$ become undefined
- the values $c_i, i > k$ increase by 1.

Each marker will only be injured finitely many times. We let $c_i[s]$ denote the value of c_i at stage s.

At each stage s let $t_i[s]$ be the least number t such that $K_{N_i}(A \restriction_t)[s] > K(t)[s] + c_i[s]$. Each marker m_i has the incentive to move at some stage $s+1$ if it observes a set of descriptions of segments of $A[s]$ that are longer than its current position, of sufficient weight. This weight is determined by the threshold $q_i[s]$ and is set to $2^{-K(t_i)[s]-c_i[s]}$. The marker m_i requires attention at stage $s+1$ if $m_i[s]$ is defined, $m_i[s] \notin A[s]$ and one of the following occurs:

(a) $i \in \emptyset'[s+1]$;
(b) $\sum_{m_i[s]<j\leq s} 2^{-K(A\restriction_j)[s]} \geq q_i[s]$;

For each $i \in \mathbb{N}$ we set $c_i[0] = i + 3$. At each stage $s+1$ the machines N_i will be adjusted according to changes of $K(n)$ for $n < t_i[s]$. This is done by running the following subroutine.

(3.2) For each $i \leq s$ and each $n < t_i[s]$, if $K(n)[s+1] < K(n)[s]$ then enumerate an N_i-description of $A[s] \restriction_n$ of length $K(n)[s+1] + c_i[s]$.

A *large* number at stage $s+1$ is one that is larger than any number that has been the value of any parameter in the construction up to stage s.

3.1. **Construction of B, M, N_i.** At stage 0 place m_0 on 0. At stage $s+1$ run (3.2). If none of the currently defined markers requires attention, let k be the largest number with $m_k[s]\downarrow$, let n be the least number $< s$ such that $K_M(B \restriction_n)[s] > K(A \restriction_n)[s]$ and

- place m_{k+1} on the least *large* number;

- enumerate an M-description of $B[s]\restriction_n$ of length $K(A\restriction_n)[s]$.

Otherwise let n be the least number such that m_n requires attention, put $m_n[s]$ into B, let $m_n[s+1]$ be a *large* number and for each $k < s$ such that $k > m_n[s]$ and $K_M(B\restriction_k)[s] \leq K(A\restriction_k)[s]$ enumerate an M-description of $B[s]\restriction_k$ of length $K(A\restriction_k)[s]$. Moreover declare $m_i[s+1]$, $i > n$ undefined, set $c_j[s+1] = c_j[s]+1$ for each $j > n$ and if (b) applies, enumerate an N_i-description of $A\restriction_{t_i}[s]$ of length $K(t_i)[s]$.

3.2. **Verification.** When m_i is first defined at some stage s it takes a *large* value so $t_i[s] < m_i[s]$. Moreover t_i can only increase when N_i computations are enumerated on strings of length t_i, which happens only when m_i moves. Hence by induction we have (3.3).

$$\text{For all } i, s, \text{ if } m_i[s] \text{ is defined then } t_i[s] < m_i[s]. \tag{3.3}$$

If $K(n)$ decreases at some stage $s+1$ for some $n < t_i[s]$, subroutine (3.2) will ensure that $K_{N_i}(A\restriction_n)[s+1] \leq K(n)[s+1]+c_i[s+1]$. Hence t_i may only decrease at $s+1$ if $A[s+1]\restriction_{t_i[s]} \neq A[s]\restriction_{t_i[s]}$, which implies (3.4).

$$\text{If } A[s]\restriction_{t_i[s]} = A[s+1]\restriction_{t_i[s]} \text{ then } t_i[s] \leq t_i[s+1]. \tag{3.4}$$

The enumeration of descriptions into N_i occurs with overhead c_i, in the sense that at stage s any description of a string of length n that is defined in N_i has length $K(n)[s]+c_i[s]$. This implies (3.5).

$$\begin{array}{l}\text{At any stage } s \text{ the weight of the } N_i\text{-descriptions that}\\ \text{describe initial segments of } A[s] \text{ is less than } 2^{-c_i[s]}.\end{array} \tag{3.5}$$

For each i there is a machine N_i as prescribed in the construction.

Lemma 3.1. *For each i the weight of the requests in N_i is bounded.*

Proof. The weight of the requests that are enumerated in N_i by subroutine (3.2) is bounded by the weight of the domain of U, which is at most 2^{-2}. Every time request is enumerated into N_i due to marker m_i requiring attention at some stage $s+1$, the marker moves to a *large* value and the weight of the request is $q_i[s] \leq \sum_{m_i[s]<j\leq s} 2^{-K(A\restriction_j)[s]}$. Hence by induction the weight of the requests that are enumerated in N_i in this way is also bounded by the weight of the domain of U. Hence $\texttt{wgt}(N_i) \leq 2^{-2}+2^{-2} = 2^{-1}$. □

Each description that is enumerated in M corresponds to a unique description in the domain of the universal machine U of the same length. Indeed, when the construction requests M to describe some $B[s]\restriction_x$ at stage $s+1$, this is in order to achieve $K_M(B\restriction_x)[s] \leq K(A\restriction_x)[s]$. Hence the new description in M corresponds to the (least) shortest description in U of $A\restriction_x[s]$. This correspondence is not one-to-one. Since the approximation to B changes in the course of the construction, different descriptions enumerated in M may correspond to the same description in the domain of U. If a U-description σ corresponds to n distinct M-descriptions we say that σ is used n times.

Let S_0 be the domain of U and for each $k > 0$ let S_k contain the descriptions in the domain of U which are used at least k times. Note that $S_{i+1} \subseteq S_i$ for each i. According to the correspondence between the domains of U, M a string σ in the

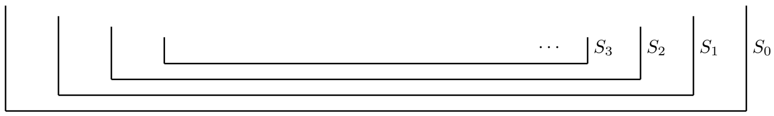

Figure 1. Infinite nested decanter model.

domain of U that is used k times incurs weight $k \cdot 2^{-|\sigma|}$ to the domain of M. Hence (3.6).

$$\texttt{wgt}(M) \leq \sum_k k \cdot \texttt{wgt}(S_k). \tag{3.6}$$

A U-description is called *active* at stage s if $U(\sigma)[s] \subseteq A[s]$. By the construction, all descriptions that enter S_1 at some stage s are currently active. More generally, only currently active strings may move from S_k to S_{k+1} at any given stage.

The sets S_k may be visualized as the nested containers of the infinite decanter model of Figure 1. As the figure indicates, descriptions may move from S_k to S_{k+1} but they also remain in S_k. If at some stage a marker m_i moves (while $m_j, j < i$ remain stable) some strings move from S_k to S_{k+1} for various $k \in \mathbb{N}$ (i.e. they are used one more time). In this case we say that these strings where reused by m_i. In order to calculate a suitable upper bound for each $\texttt{wgt}(S_k)$ we need (3.7).

(3.7) If during the interval of stages $[s, r]$ a marker m_n is not injured and $n \notin \emptyset'[r]$ then the weight of the strings that m_n reuses during this interval which remain active at stage r is at most $2^{-c_n[s]}$.

Indeed, by clause (b) each time m_n reuses some U-descriptions at some stage $x+1$, the weight of these descriptions is at most $q_n[x]$. Moreover when such an event happens, a string of weight $q_n[x]$ is used to describe $A \restriction_{t_i} [x]$ via N_n. By (3.3) if at least one of the descriptions in U that m_n repaid at some stage $x \in [s, r]$ continues to be active at stage r, then $A[x] \restriction_{t_n[s]} = A[r] \restriction_{t_n[s]}$. Hence by (3.4) we get that $t_n[y] \geq t_n[x]$ for all $y \in [x, r]$. So during the stages in $[x, r]$ the weight of the descriptions in U that m_n repaid and remain active at stage r is bounded by the weight of the descriptions in N_n that describe segments of $A[r]$. By (3.5) this is at most $2^{-c_n[s]}$. This concludes the proof of (3.7).

Lemma 3.2. *The weight of the requests that are enumerated in M is finite.*

Proof. Since only strings in the domain of U are used, $\texttt{wgt}(S_1) < 2^{-2}$ and $\texttt{wgt}(S_2) < 2^{-2}$. Let $k > 1$. Every entry of a string into S_{k+1} is due to a marker m_x which reused it when it was already in S_k. Since $k > 1$, this string entered S_k due to another marker m_y with $y > x \geq 0$. Inductively, that string entered S_1 due to a marker m_z with $z \geq k-1$. Fix z, let S_k^z contain the strings in S_k that entered S_1 due to marker m_z and let (s_i) be the increasing sequence of stages where m_z is injured. Note that at this point we do not assume that (s_j) is a finite. Since the movement of a marker m_i injures all $m_j, j > i$, the only stages where strings move from S_k^z to S_{k+1} are the (s_i). Moreover since only active strings move from S_k^z to S_{k+1} at stage s_i, according to (3.7) their weight is bounded by $2^{-c_z[s_i-1]}$. So the weight of the strings that enter S_{k+1} from S_k^z is bounded by $\sum_j 2^{-c_z[s_j-1]}$. Since $c_z[s_{j+1}-1] = c_z[s_j] < c_z[s_j-1]$ for all j, this weight is bounded by $\sum_j 2^{-c_z[0]-j} = 2^{-c_z[0]+1}$. Since $c_z[0] = z+3$ this bound becomes 2^{-z-2}. Since $S_k = \cup_{z \geq k-1} S_k^z$ the total weight of the strings

that enter S_{k+1} from S_k is bounded by $\sum_{z\geq k-1} 2^{-z-2} = 2^{-k}$. Therefore by (3.6) the weight of M is finite. $\square$

By Lemma 3.2 the machine M prescribed in the construction exists and (3.1) is met. We conclude with the proof that $\emptyset' \leq_T B$. The proof of the following Lemma uses the fact that each N_i is a prefix-free machine, which was established in Lemma 3.1.

Lemma 3.3. *Each marker m_i is injured only finitely many times and it reaches a limit.*

Proof. Assume that this holds for all $i < k$. Then m_k stops being injured after some stage s_0. Hence c_k reaches a limit at s_0. Since A is not K-trivial there is some least n such that $K_{N_k}(A\restriction_n) > K(n)+c_k$. If s_1 is a stage where the approximations to $A\restriction_n$ and $K(i)$, $i \leq n$ have settled then the approximations to t_k, q_k also reach a limit by this stage. If marker m_k moved after stage s_1 the construction would enumerate an N_k-description of $A\restriction_n$ of length $K(n)+c_k[s_0]$ which contradicts the choice of n. Hence m_k reaches a limit by stage s_1 and this concludes the induction step. $\square$

By Lemma 3.3 and the construction we get that the movement of the markers satisfies properties (i)-(v) of Section 2. Hence $\emptyset' \leq_T B$, which concludes the verification of the construction and the proof of Theorem 1.1.

4. Proof of Theorem 1.2

Let A, D be two computably enumerable sets which are not K-trivial. For the proof of Theorem 1.2 it suffices to construct a computably enumerable set B such that $B \leq_K A$, $B \leq_K D$ and $B \equiv_T \emptyset'$. This follows from the downward density of the c.e. K-degrees as discussed in Section 3. The coding of $\emptyset'$ into B will be done via the markers (m_i) and the relations $B \leq_K A$, $B \leq_K D$ will be achieved with the construction of two prefix-free machines M_a, M_d respectively such that

$$K_{M_a}(B\restriction_n) \leq K(A\restriction_n) \quad \text{and} \quad K_{M_d}(B\restriction_n) \leq K(D\restriction_n) \quad \text{for all } n. \tag{4.1}$$

4.1. **Merging two constructions.** The basic plan of the construction of M_a, M_d is to merge a construction for M_a of the type that was given in Section 3 with a construction for M_d of the same type. Note that we will have a single set of markers m_i but their movement will be stimulated by both requirements in (4.1). We will use the same set of constants c_i for both A and D, since their values only depend on the movement of the markers on B. However for each i we have N_i^a, N_i^d instead of N_i. Moreover at each stage s we let $t_i^a[s]$ be the least number x such that $K_{N_i^a}(A\restriction_x)[s] > K(x)[s] + c_i[s]$ and we let $t_i^d[s]$ be the least number y such that $K_{N_i^d}(D\restriction_y)[s] > K(y)[s] + c_i[s]$. For each $i \in \mathbb{N}$ we set $c_i[0] = i + 4$. The universal machine U and the notion of injury of a marker remains the same. In particular, if at some stage s marker m_k moves while m_j, $j < k$ remain constant this causes m_i, $i > k$ to be injured. This means that m_i, $i > k$ become undefined and the values of $c_i, i > k$ increase by 1.

At each stage $s+1$ the machines N_i^a, N_i^d will be adjusted according to changes of $K(n)$ for $n < t_i[s]$. This is done by running the following subroutine.

(4.2) For each $i \leq s$, if $K(n)[s+1] < K(n)[s]$ for some $n < t_i^a[s]$ or $n < t_i^d[s]$ then enumerate an N_i^a-description of $A[s] \upharpoonright_n$ of length $K(n)[s+1] + c_i[s]$ or an N_i^d-description of $D[s] \upharpoonright_n$ of length $K(n)[s+1] + c_i[s]$ respectively.

The thresholds $q_i^a[s]$, $q_i^d[s]$ are set to $2^{-K(t_i^a)[s]-c_i[s]}$, $2^{-K(t_i^d)[s]-c_i[s]}$ respectively and play a similar role as $q_i[s]$ in the argument of Section 3. However since the complexities $K(A \upharpoonright_n), K(D \upharpoonright_n)$ may differ for various n, the definition of a marker requiring attention will be modified, as we elaborate in Section 4.2.

4.2. **Lack of uniformity and solution.** The main issue that we have to deal with when we merge two constructions of the type used in Section 3 which depend on different c.e. sets A, D is that the thresholds q_i^a, q_i^d that correspond to some marker m_i may have different values. Hence the marker may be motivated to move by M_a but not by M_b. This lack of uniformity has an impact in the calculations of the weight of the machines N_i^a, N_i^d which affects a verification on the lines of Section 3.2.

The solution is to use the additional parameters p_i^a, p_i^d which record a value of q_i^d and q_i^a respectively which motivates marker m_i to move at some stage. In particular, at some stage $s+1$ we may have $\sum_{m_i[s]<j\leq s} 2^{-K(A\upharpoonright_j)[s]} \geq q_i^a[s]$ but this may not hold for D in place of A and $q_i^d[s]$ in place of $q_i^d[s]$. This means that at this stage M_a requires the movement of m_i but M_d does not. At such a stage we will move m_i for the sake of M_a, also enumerating an N_i^a-description of $t_i^a[s]$ of length $K(t_i^a)[s] + c_i[s]$. However an enumeration of an N_i^d-description of $t_i^d[s]$ of length $K(t_i^d)[s] + c_i[s]$ is not justified and will not take place. Instead, we will store the value of $q_i^a[s]$ into $p_i^d[s+1]$. Since $q_i^a[s]$ is an upper bound to the M_d descriptions we need to pay due to the movement of m_i at stage $s+1$, at the next stage the threshold in condition for the movement of m_i for the sake of D will be $q_i^d[s+1] - p_i^d[s+1]$. As long as m_i moves for the sake of M_a the value of p_i^d will keep on increasing, recording upper bounds to the M_d descriptions that we need to pay due to the M_a-motivated movements of m_i. When m_i moves for the sake of M_d, the value of p_i^d will drop to 0 and the enumeration into N_i^d will be justified. The same holds symmetrically for A with q_i^a and p_i^a. With this amendment a combined construction can be verified along the lines of the argument of Section 3.2.

According to the above motivation, we say that the marker m_i requires attention at stage $s+1$ if $m_i[s]$ is defined, $m_i[s] \notin A[s]$ and one of the following occurs:

(a) $i \in \emptyset'[s+1]$;
(b) $\sum_{m_i[s]<j\leq s} 2^{-K(A\upharpoonright_j)[s]} \geq q_i^a[s] - p_i^a[s]$;
(c) $\sum_{m_i[s]<j\leq s} 2^{-K(D\upharpoonright_j)[s]} \geq q_i^d[s] - p_i^d[s]$;

The definition of a *large* is as in the argument of Section 3.

4.3. **Construction of $B, M_a, M_d, N_i^a, N_i^d$.** At stage 0 place m_0 on 0. At stage $s+1$ run (4.2). If none of the currently defined markers requires attention, let k be the largest number such that $m_k[s]\downarrow$, let n_a be the least number that is less than s such that $K_{M_d}(B \upharpoonright_{n_a})[s] > K(A \upharpoonright_{n_a})[s]$, let n_d be the least number that is less than s such that $K_{M_d}(B \upharpoonright_{n_d})[s] > K(D \upharpoonright_{n_d})[s]$ and

- place m_{k+1} on the least *large* number;
- enumerate an M_a-description of $B[s] \upharpoonright_{n_a}$ of length $K(A \upharpoonright_{n_a})[s]$ and an M_d-description of $B[s] \upharpoonright_{n_d}$ of length $K(D \upharpoonright_{n_d})[s]$.

Otherwise let n be the least number such that m_n requires attention, put $m_n[s]$ into B, let $m_n[s+1]$ be a *large* number and for each $k < s$ with $k > m_n[s]$

- if $K_{M_a}(B \upharpoonright_k)[s] \leq K(A \upharpoonright_k)[s]$ enumerate an M_a description of $B[s+1] \upharpoonright_k$ of length $K(A \upharpoonright_k)[s]$.
- if $K_{M_d}(B \upharpoonright_k)[s] \leq K(D \upharpoonright_k)[s]$ enumerate an M_d-description of $B[s+1] \upharpoonright_k$ of length $K(D \upharpoonright_k)[s]$.

Moreover declare $m_i[s+1]$, $i > n$ undefined and set $c_j[s+1] = c_j[s]+1, p_j^a[s+1] = p_j^d[s+1] = 0$ for each $j > n$. If both (b), (c) hold, enumerate an N_i^a-description of $A \upharpoonright_{t_i^a} [s]$ of length $K(t_i^a)[s]$, enumerate an N_i^d-description of $D \upharpoonright_{t_i^d} [s]$ of length $K(t_i^d)[s]$ and set $p_i^a[s+1] = p_i^d[s+1] = 0$. Otherwise, if (b) holds enumerate an N_i^a-description of $A \upharpoonright_{t_i^a} [s]$ of length $K(t_i^a)[s]$, set $p_i^a[s+1] = 0$ and $p_i^d[s+1] = p_i^d[s+1] + q_i^a[s]$. Otherwise, if (a) holds enumerate an N_i^d-description of $A \upharpoonright_{t_i^d} [s]$ of length $K(t_i^d)[s]$, set $p_i^d[s+1] = 0$ and $p_i^a[s+1] = p_i^a[s+1] + q_i^d[s]$.

4.4. **Verification.** As in the verification of Section 3 we have (4.3).

$$\text{For all } i, s, \text{ if } m_i[s] \text{ is defined then } t_i^a[s] < m_i[s] \text{ and } t_i^d[s] < m_i[s]. \tag{4.3}$$

Moreover the justification of (3.4) also applies to (4.4).

$$\begin{aligned} &\text{If } A[s] \upharpoonright_{t_i^a[s]} = A[s+1] \upharpoonright_{t_i^a[s]} \text{ then } t_i^a[s] \leq t_i^a[s+1]. \\ &\text{If } D[s] \upharpoonright_{t_i^d[s]} = D[s+1] \upharpoonright_{t_i^d[s]} \text{ then } t_i^d[s] \leq t_i^d[s+1]. \end{aligned} \tag{4.4}$$

The same holds for the analogue (4.5) of (3.5).

(4.5) At any stage s the weight of the N_i^a-descriptions that describe initial segments of $A[s]$ and the weight of the N_i^d-descriptions that describe initial segments of $D[s]$ are both less than $2^{-c_i[s]}$.

For each i there are machines N_i^a, N_i^d as prescribed in the construction. The proof of this fact is slightly more involved than the corresponding fact in the argument of Section 3 due to the amendment that was discussed in Section 4.2.

Lemma 4.1. *For each i the weights of the requests in N_i^a and N_i^d are bounded.*

Proof. The weight of the requests that are enumerated in N_i^a by subroutine (4.2) at the beginning of each stage is bounded by the weight of the domain of U, which is at most 2^{-2}. The same applies to N_i^d.

Let (s_j) be the sequence of stages where m_i moves, inside an interval J of stages s where m_i is not injured and $i \not\in \emptyset'[s]$. Moreover let $I_j = (m_i[s_j], m_i[s_{j+1}]]$ be the interval that marker m_i crosses when it moves at stage s_j. For each j let

$$a_j = \sum_{n \in I_j} 2^{-K(A \upharpoonright_n)[s]} \qquad \text{and} \qquad d_j = \sum_{n \in I_j} 2^{-K(D \upharpoonright_n)[s]}.$$

Since $i \not\in \emptyset'[s_j]$ for all j, each time marker m_i moves during these stages, it requires attention by (b) or (c). Hence for each j exactly one of the following holds:

- a request of weight at most $a_j - p_i^a[s_j - 1]$ is enumerated in N_i^a;
- p_i^a increases by at most d_j.

Consequently, if $s_{j_0} < s_{j_1} < \cdots$ are the stages in J where N_i^a enumerations occur, the weight of the enumeration at stage s_{j_n} is at most $\sum_{j_{n-1}<k<j_n} d_k + a_{j_n}$. If we let $y_j = \max\{a_j, d_j\}$ then this bound becomes $\sum_{j_{n-1}<k\le j_n} y_k$. If we iterate this argument for each maximal interval J of stages s where m_i is not injured and $i \notin \emptyset'[s]$ the corresponding intervals I_j will be disjoint, because each time m_i is injured it is redefined to be a *large* number. Moreover the possible enumeration of i into $\emptyset'[s]$ does not cause any enumeration into N_i^a, it happens at most once and causes m_i to reach a limit. If we let w_n to be the weight of the shortest U-description of any string of length n, we have $w_n \ge 2^{-K(A\restriction_n)[s]}$ and $w_n \ge 2^{-K(D\restriction_n)[s]}$ for each stage s. Hence the total weight that is enumerated in N_i^a due to movements of m_i during the construction is bounded by $\sum_n w_n < 2^{-2}$. If we combine this with the weight that is added by applications of (4.2) we get $\texttt{wgt}(N_i^a) < 2^{-1}$. The same argument applies to N_i^d symmetrically, giving $\texttt{wgt}(N_i^d) < 2^{-1}$. $\square$

As in the argument of Section 3, there is a many-one correspondence between the domain of M_a and the domain of the universal machine U. We say that a U-description is A-*used* if it corresponds to a string in the domain of M_a. Moreover it is A-*used* n times if it corresponds to n different strings in the domain of M_a. Let S_0^a be the domain of U and for each $k > 0$ let S_k^a contain the descriptions in the domain of U which are A-used at least k times. Note that $S_{i+1}^a \subseteq S_i^a$ for each i. According to the correspondence between the domains of U and M^a, a string σ in the domain of U that is A-used k times incurs weight $k \cdot 2^{-|\sigma|}$ to the domain of M^a. Similar terminology and observations apply on D and M_d. Hence we have (4.6).

$$\texttt{wgt}(M_a) \le \sum_k k \cdot \texttt{wgt}(S_k^a) \quad \text{and} \quad \texttt{wgt}(M_d) \le \sum_k k \cdot \texttt{wgt}(S_k^d). \tag{4.6}$$

A U-description is called A-*active* at stage s if $U(\sigma)[s] \subseteq A[s]$. By the construction, only currently A-active strings may move from S_k^a to S_{k+1}^a and only currently D-active strings may move from S_k^d to S_{k+1}^d at any given stage.

The sets S_k^a and S_k^d may be visualized as the containers of two independent decanter models that are identical to the one illustrated in Figure 1. If at some stage a marker m_i moves (while $m_j, j < i$ remain stable) some strings move from S_k^a to S_{k+1}^a and from S_k^d to S_{k+1}^d for various $k \in \mathbb{N}$. In this case we say that these strings where A-reused and D-reused respectively by m_i.

The justification of the following is slightly more complex than the analogous (3.7) of Section 3 due to the amendment that was discussed in Section 4.2.

(4.7) If during the interval of stages $[s, r]$ a marker m_n is not injured and $n \notin \emptyset'[r]$ then the weight of the strings that are A-reused by m_n during this interval which remain active at stage r is at most $2^{-c_n[r]} + p_n^a[r]$.

We follow the proof of (3.7), where parameters q_n, t_i are replaced with q_n^a, t_i^a and facts (3.3), (3.4), (3.5) are replaced by (4.3), (4.4), (4.5) respectively. This argument applies only on the stages where the movement of m_i is accompanied by an enumeration into N_i^a. For the rest of the stages the increase of p_n^a covers the increase in the weight of the strings that are A-reused. At the stages $k+1$ where N_i^a enumerations occur, $p_n^a[k+1]$ becomes 0 but the bound $2^{-c_n[k]}$ is reinstated since the weight of the N_i^a enumeration is at least $p_n^a[k]$ plus the weight of the strings that are A-reused at $k+1$. This gives the bound $2^{-c_n[r]} + p_n^a[r]$ during the interval.

The same argument applies symmetrically to the strings that are D-used, providing the bound $2^{-c_n[s]} + p^d_n[s]$.

$$\text{If during the interval of stages } [s,r] \text{ a marker } m_n \text{ is not injured then the weight of the strings that are } D\text{-reused by } m_n \text{ during this interval which remain active at stage } r \text{ is at most } 2^{-c_n[s]} + p^d_n[s]. \tag{4.8}$$

Note that $p^a_n[s] \le q^a_n[s]$ for each n and all stages s. This follows from clause (b) in Section 4.2 and the fact that whenever m_n moves due to this clause (or is injured) parameter p^a_n takes value 0. On the other hand by the definition of q^a_n we have $q^a_n[s] < 2^{-c_n[s]}$, so $p^a_n[s] \le 2^{-c_n[s]}$. Hence the bound in (4.7) can be replaced with $2^{-c_n[s]+1}$. A similar argument applies to $p^d_n[s]$. The proof of Lemma 4.2 uses this observation in an adaptation of the proof of the analogous Lemma 3.2.

Lemma 4.2. *The weight of the requests that are enumerated in M_a is finite; the same holds for M_d.*

Proof. We give the proof for M_a; the proof for M_d is entirely symmetric. Since only strings in the domain of U are A-used, $\texttt{wgt}(S^a_1) < 2^{-2}$ and $\texttt{wgt}(S^a_2) < 2^{-2}$. Let $k > 1$. Every entry of a string into S^a_{k+1} is due to a marker m_x which A-reused it when it was already in S^a_k. Since $k > 1$, this string entered S^a_k due to another marker m_y with $y > x \ge 0$. Inductively, that string entered S^a_1 due to a marker m_z with $z \ge k-1$. Fix z, let $S^a_k(z)$ contain the strings in S^a_k that entered S^a_1 due to marker z and let (s_i) be the increasing sequence of stages where m_z is injured. Since the movement of a marker m_i injures all $m_j, j > i$, the only stages where strings move from $S^a_k(z)$ to S^a_{k+1} are the (s_i). Moreover since only A-active strings move from $S^a_k(z)$ to S^a_{k+1} at stage s_i, according to (4.7) their weight is bounded by $2^{-c_z[s_i-1]} + p^a_z[s_i-1]$. Since $p^a_z[s] \le 2^{-c_z[s]}$ the latter is bounded by $2^{-c_z[s_i-1]+1}$. So the weight of the strings that enter S^a_{k+1} from $S^a_k(z)$ is bounded by $\sum_j 2^{-c_z[s_j-1]+1}$. Since $c_z[s_{j+1}-1] = c_z[s_j] < c_z[s_j-1]$ for all j, this weight is bounded by $\sum_j 2^{-c_z[0]-j+1} = 2^{-c_z[0]+2}$. Since $c_z[0] = z+4$ this bound becomes 2^{-z-2}. Since $S^a_k = \cup_{z\ge k} S^a_k(z)$, the total weight of the strings that enter S^a_{k+1} from S^a_k is bounded by $\sum_{z\ge k-1} 2^{-z-2} = 2^{-k}$. Therefore by (4.6) the weight of M_a is finite. □

By Lemma 4.2 the machines M_a, M_d prescribed in the construction exist and (4.1) is met. We conclude with a proof that $\emptyset' \le_T B$. This is based on an argument that the markers m_i converge, and is slightly more involved than the proof of the analogous Lemma 3.3 since in the combined construction the markers move for the sake of both M_a and M_d.

Lemma 4.3. *Each marker m_i is injured only finitely many times and it reaches a limit.*

Proof. Assume that this holds for all $i < k$. Then m_k stops being injured after some stage s_0. Hence $c_k[s]$ reaches a limit c_k at s_0. If $k \in \emptyset'$ the marker m_k will stop requiring attention as well as moving after s_0 and after this enumeration has taken place. So assume that $k \notin \emptyset'$. Since A is not K-trivial there is some least n_a such that $K_{N_k}(A\restriction_{n_a}) > K(n_a)+c_k$. If $s_1 > s_0$ is a stage where the approximations to $A\restriction_n$ and $K(i)$, $i \le n$ have settled then the approximations to t^a_k, q^a_k also reach a limit by this stage. The limit of t^a_k is n_a. The same observations apply to D and show that there is some stage $s_2 > s_1$ by which the parameters t^d_k, q^d_k reach

a limit. Let n_d be the limit of t_k^d. If marker m_k moved after stage s_2, this would be either due to clause (b) or due to clause (c) of Section 4.2. In the first case the construction would enumerate an N_k^a-description of $A\restriction_{n_a}$ of length $K(n_a)+c_k[s_0]$ and in the second case an N_k^d-description of $D\restriction_{n_d}$ of length $K(n_d)+c_k[s_0]$. The first action would contradict the choice of n_a and the second action would contradict the choice of n_d. Hence m_k reaches a limit by stage s_2 and this concludes the induction step. $\square$

By Lemma 4.3 and the construction we get that the movement of the markers satisfies properties (i)-(v) of Section 2. Hence $\emptyset' \leq_T B$, which concludes the verification of the construction and the proof of Theorem 1.2.

References

[Bar05] George Barmpalias. Computably enumerable sets in the Solovay and the strong weak truth table degrees. In S. Barry Cooper, Benedikt Löwe, and Leen Torenvliet, editors, *CiE*, volume 3526 of *Lecture Notes in Computer Science*, pages 8–17. Springer, 2005.

[Bar10a] George Barmpalias. Elementary differences between the degrees of unsolvability and the degrees of compressibility. *Ann. Pure Appl. Logic*, 161(7):923–934, 2010.

[Bar10b] George Barmpalias. Relative randomness and cardinality. *Notre Dame J. Formal Logic*, 51(2), 2010.

[Bar11] George Barmpalias. On strings with trivial Kolmogorov complexity. *Int J Software Informatics*, 5(4):609–623, 2011.

[BB10] Martijn Baartse and George Barmpalias. On the gap between trivial and nontrivial initial segment prefix-free complexity. Submitted, 2010.

[BM09] George Barmpalias and Anthony Morphett. Non-cupping, measure and computably enumerable splittings. *Mathematical Structures in Computer Science*, 19(1):25–43, 2009.

[BMN11] Laurent Bienvenu, Wolfgang Merkle, and André Nies. Solovay functions and K-triviality. In *STACS*, pages 452–463, 2011.

[BV11] George Barmpalias and C.S. Vlek. Kolmogorov complexity of initial segments of sequences and arithmetical definability. *Theoretical Computer Science*, 412(41):5656 – 5667, 2011.

[Cha75] Gregory J. Chaitin. A theory of program size formally identical to information theory. *J. Assoc. Comput. Mach.*, 22:329–340, 1975.

[CM06] Barbara F. Csima and Antonio Montalbán. A minimal pair of K-degrees. *Proc. Amer. Math. Soc.*, 134(5):1499–1502 (electronic), 2006.

[DH10] Rod Downey and Denis Hirshfeldt. *Algorithmic Randomness and Complexity.* Springer, 2010.

[DHL04] Rod G. Downey, Denis R. Hirschfeldt, and Geoff LaForte. Randomness and reducibility. *J. Comput. System Sci.*, 68(1):96–114, 2004.

[DHL07] Rodney G. Downey, Denis R. Hirschfeldt, and Geoffrey LaForte. Undecidability of the structure of the solovay degrees of c.e. reals. *J. Comput. Syst. Sci.*, 73(5):769–787, 2007.

[DHN02] R. Downey, D. Hirschfeldt, and A. Nies. Randomness, computability and density. *SIAM J. Computing*, 31:1169–1183, 2002.

[DHNS03] Rod G. Downey, Denis R. Hirschfeldt, André Nies, and Frank Stephan. Trivial reals. In *Proceedings of the 7th and 8th Asian Logic Conferences*, pages 103–131, Singapore, 2003. Singapore Univ. Press.

[KHMS12] Bjørn Kjos-Hanssen, Joseph S. Miller, and Reed Solomon. Lowness notions, measure and domination. *J. Lond. Math. Soc.*, 2012. In press.

[Kol65] A. N. Kolmogorov. Three approaches to the definition of the concept "quantity of information". *Problemy Peredači Informacii*, 1(vyp. 1):3–11, 1965.

[Lev73] L. A. Levin. The concept of a random sequence. *Dokl. Akad. Nauk SSSR*, 212:548–550, 1973.

[ML66] Per Martin-Löf. The definition of random sequences. *Information and Control*, 9:602–619, 1966.

[MS07] Wolfgang Merkle and Frank Stephan. On C-degrees, H-degrees and T-degrees. In *Twenty-Second Annual IEEE Conference on Computational Complexity (CCC 2007), San Diego, USA, 12–16 June 2007*, pages 60–69, Los Alamitos, CA, USA, 2007. IEEE Computer Society.

[MY08] Joseph S. Miller and Liang Yu. On initial segment complexity and degrees of randomness. *Trans. Amer. Math. Soc.*, 360(6):31933210, 2008.

[MY10] Joseph S. Miller and Liang Yu. Oscillation in the initial segment complexity of random reals. *Adv. Math.*, 2010. In press.

[Nie05] André Nies. Lowness properties and randomness. *Adv. Math.*, 197(1):274–305, 2005.

[Nie09] André Nies. *Computability and Randomness*. Oxford University Press, 2009.

[Sol75] R. Solovay. Handwritten manuscript related to Chaitin's work. IBM Thomas J. Watson Research Center, Yorktown Heights, NY, 215 pages, 1975.

[YDD04] Liang Yu, Decheng Ding, and Rod Downey. The Kolmogorov complexity of the random reals. *Ann. Pure Appl. Logic*, 129(1–3):163–180, 2004.

George Barmpalias: State Key Laboratory of Computer Science, Institute of Software, Chinese Academy of Sciences, Beijing 100190, P.O. Box 8718, China.

E-mail address: barmpalias@gmail.com

URL: http://www.barmpalias.net